Potentialfinder - Fostering Network Innovation by Connecting Data Owners Using Scaled Business-Relevant Pattern Recognition and Clustering

Mathias Riechert
mathias.riechert@gmail.com

Roshan Bhandari
Clemson University
rbhanda@clemson.edu

Abhijeet Amle
Clemson University
aamle@clemson.edu

Abhimanyu Abhinav
Clemson University
aabhina@clemson.edu

Nina Hubig
Clemson University
nhubig@clemson.edu

Abstract

The advancement in collection, computing and storage technologies has led to an exponential growth of available data in multiple disciplines. However, the human capacity of analyzing this data does not grow at the same rate, leaving a vast amount of potential disparate, invisible and unused. We want to enhance the capability of humans to automatically find relevant patterns in data to leverage potential in this increasing sea of data. We present an innovation network creation framework and Python library that detects exponential growth patterns from publicly available tabular data. It works as a magnifying glass to reveal the most relevant parts of the data and the processes represented by it. The extracted exponential patterns can be useful for topic or disease detection as well as for organisations such as venture capital and consulting firms to improve investment decisions. Additionally, startups and innovation units in corporates can leverage these information to base their business models on insights into sectors, markets or customer segments with exponential growth. To foster the innovation based on the revealed patterns, we connect the respective data owners that uploaded similar patterns. This paper proposes a framework for networked innovation creation including a) an algorithm to automatically detect exponential, b) approaches to scale its application to public tabular data in different sizes and formats, c) a similarity network connecting the found patterns to innovation networks, d) a clustering to group the data owners and enable co- and crowd innovation. We run experiments on large scale data for all steps to provide evidence for cost-efficiency, scalability and feasibility of the contributions.

1. Introduction

With the emergence of digital technologies, the amount of data collected, generated and stored world-wide grows exponentially. Private and public organizations are storing petabytes of data in their repositories. The human and organizational capacity to analyze those data does not scale at the same rate, resulting in business and humanitarian potential being left unused. In this study we demonstrate the application of our innovation network framework and algorithms on data from the platform Kaggle, which serves as a central hub of publicly available data. The same framework and approach can be applied to data lakes and repositories in private organizations as well to connect their innovation and business units around similar patterns. In this research, we focus our detection on exponential patterns over time, because they represent processes and developments of exponential growth, where minor improvements or investments can have a huge impact, yielding the highest potential return on invested time.

The extracted exponential patterns can be useful for topic or disease detection as well as for organisations such as venture capital and consulting firms to improve investment decisions. Additionally startups can leverage the patterns found to base their business models on insights into sectors, markets or customer segments with exponential or logistic growth. However, the data that needs to be processed and mined is of high volume and comes in various formats and structure, increasing flexibility and robustness requirements of general automated analyses like the one we demonstrate.

Both in public and private organizations, Porter's [1] view of the value chain flowing from upstream suppliers to the end customer is increasingly giving way to network models to creating and delivering value [2][3][4][5]. The term "Network Innovation" refers to innovation taking place in networks of people and organizations. Lyytinen, Yoo, and Boland Jr. [3] also view innovation emerging from networks as a flow of fragile, heterogeneous and dynamic knowledge in 'trading zones' [6] shared between heterogeneous actors and their tools in complex socio-technical networks.

We present a framework for finding business-relevant patterns in data and sparking innovation networks and communities around them:

URI: https://hdl.handle.net/10125/79509
978-0-9981331-5-7

- **Contribution 1:** Development and evaluation of the algorithm 'Potential Finder' that can be run in any commodity machine to find exponential (potentials) and logistic patterns in any data.
- **Contribution 2:** Development and evaluation of an architecture that can scale contribution 1 to terabytes of data to find exponential patterns.
- **Contribution 3:** Creation of a similarity network between all found patterns, and creation of a data owner network based on their uploaded patterns.
- **Contribution 4:** Clustering of innovation communities based on the pattern and network properties to organize the data owners of the respective patterns and prevent their communication overload.

2. Theoretical Background

Digital innovation is defined as "the carrying out of new combinations of digital and physical components to produce novel products" [7]. Our approach aims at identifying these new combinations of digital components by identifying business-relevant patterns in existing data and connecting responsible persons to ignite business model creation around those patterns.

Lyytinen, Yoo, Boland Jr. [3] distinguished four types of emerging innovation networks: project innovation networks (homogeneous operand resources and centralized coordination), clan innovation networks (homogeneous operand resources and distributed coordination), federated innovation networks (heterogeneous operand resources and centralized coordination) and anarchic innovation networks (heterogeneous operand resources and distributed coordination). Our framework aims to extend this emerging innovation network framework with a new framework that suggests, creates and organizes new innovation networks for anarchic and federated innovation networks by introducing pattern recognition methods from other fields. To spark innovation at the point with the highest return on invest, we suggest and foster new user groups around similar business-relevant patterns with exponential character.

Pattern Recognition has been used to identify innovation trends in science [8][9][10], medicine [11][12], finance [13], automotive [14] and many more. Innovation detection has been based on co-citation networks to identify emerging topics and trends in scientific publications [15]. Data communication costs reducing to the level of irrelevance provides motivation to broaden the detection attention above scientific communication. With rising computation power at reduced cost and exponentially growing data available, detecting innovation directly on the source data level will increasingly allow for much more immediate identification and reaction to innovation.

Exponential patterns represent processes and developments of exponential growth, where minor improvements or investments can have a huge impact, yielding the highest potential return on invested resources. Exponential pattern recognition has been used before in fields such as finance [16] and environmental studies [17]. In business, exponential growth and exponential technologies are dominant in the majority of the startup pitches and also discussed in corporate and open innovation setting [18][19].

Network theory and analysis have seen a widespread use for analyzing knowledge spread in innovation [20]. Among others, it has been used by Singh [21] to study knowledge transmission for innovation, by Fleming, Mingo, Chen [22] for Collaborative Brokerage, Generative Creativity, and Creative Success by Perry-Smith, Mannucci [23], for explaining influence drivers through the ideation phase, and for analyzing entrepreneurship and innovation by Huggins and Thompson [24]. A comprehensive review is given by Phelps, Heidl and Wadhwa [20] and more recently by Kwon, Rondi and Levin [25]. Those studies focus on analyzing and explaining innovation via knowledge spread from existing interpersonal networks. Our framework differs from the existing research in that we strive to reverse that idea by sparking collaboration communities around data patterns with business potential. Consequently, we create a network of similar patterns and use similarity clustering to suggest links between the data owners, who uploaded the data and have the context knowledge of it.

3. Research Design and Methods

We employ the Design Science paradigm, which is rooted in engineering and the sciences of the artificial [26] and develops knowledge about a problem domain and its solution in the building and application of the designed artefact [27]. We apply the synthesized Design Science paradigm [28][29] to develop an algorithm, a scalable architecture, create a similarity network and develop knowledge about the patterns found for the community. In parallel, we provide evidence for the evaluation of speed, scalability, cost and applicability on public data. To evaluate the artifacts and iteratively improve them, we conducted quantitative speed and cost experiments based on exemplary 220 GB data collected from the platform Kaggle. We are performing our research on datasets from Kaggle because they are freely

available and there is additional meta information and an uploading data owner connected. In that regard, the collection of data sets there is similar to data in data lakes and repositories found in industry. Having a uploader linked allows to validate the found patterns in a later research stage to test the pattern value in mixed-method approaches. From the Kaggle platform we downloaded 2,658 randomly selected public data bases from 794 Kaggle users. These datasets make a total of 220 GB of data and comprise .csv files, .txt files, .json files, .zip files, and images. After extracting all zip files, we removed all the files which do not have tabular data, e.g., .readme, .png, .jpg. After the data pre-processing step, we got 11,612 data tables with extensions: .csv, .tsv, and .xlsx, and the total size 180 GB which is around 81.8% of the original data. There are different separators used in the datasets, so we adapted the algorithms to work with semicolon, comma, tab, and vertical bar. Ninety percent of the data tables have a size between 10KB to 500KB, and only 27 data tables have a size greater than 1 GB, the largest file has a size of 9.7 GB. We observed large variation in the number of rows in all the files. To allow analyzing also the huge data sets, we introduce a byte-wise sampling to 5000 rows from each data set. Details of this algorithm are presented in Section 4.2.

4. Innovation Network Creation Framework

Figure 1 depicts the proposed Innovation Network Creation Framework. The cycle starts at the data platform, where the data owners have uploaded data sets in databases. The data are automatically extracted from the data platform (can be a data lake in a corporate setting; we use the public Kaggle data in this research).

In Step 1 (equals contribution 1) in Section 4.1 the algorithm for finding exponential patterns is developed. This algorithm is parallelized in Step 2 which scales this algorithm across all data sets in all databases from all data owners as described in Section 4.2. In Step 3 (Section 4.3) a similarity network is calculated to reveal connections between similar exponential patterns found in the data. This weighted network allows to connect individual data owners. To prevent a too large number of possible connections that could overwhelm the data owners, we form innovation clusters in Step 4 (Section 4.4). By being suggested a smaller subset-community around similar patterns, communication and co-innovation around similar patterns is organized in a smaller groups to keep down cognitive bias and communication overhead for each community member.

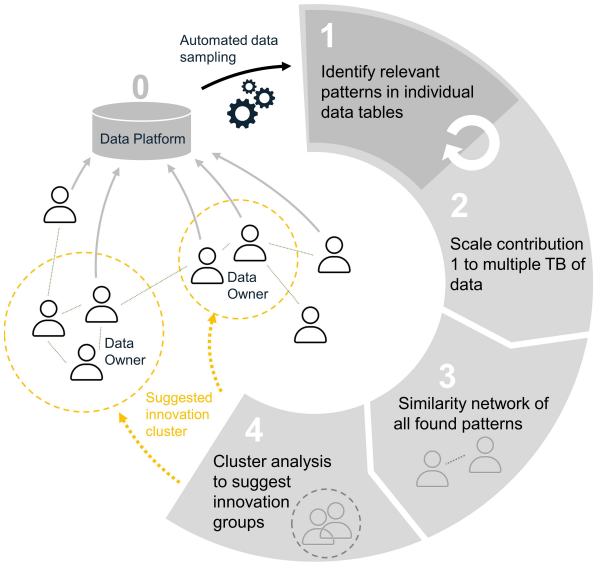

Figure 1. Innovation Network Creation Framework

In the following subsections, the artifact development and evaluation are described.

Algorithm 1 Potential Finder Algorithm

```
n = number of files
r_thresh = threshold for exponential fit
for i = 1, 2, ... n do
    Dc = list of data columns in file_i
    Dn = list of numerical column in data table
    for j = 1, 2, ... len(Dc) do
        for k = 1, 2, ... len(Dn) do
            a, b = slope of power_law_fit (Dc_j, Dn_k)
            r^2 = 1 - sum((Dn_k - a e^(b Dc_j))^2) / sum((Dn_i - mean(Dn))^2)
            if r^2 ≥ r_thresh then
                exponential pattern found
                plot_graph(Dc_j, Dn_k)
                write coefficients a,b, r^2 to result
            end if
        end for
    end for
end for
```

4.1. Pattern Finding Algorithm

As we are interested in business-relevant patterns, we focus on analyzing exponential growth over time. Consequently, Algorithm 1 first identifies all date columns in the data. Then it iterates over all combinations of date columns and numerical columns and fits an exponential function to the data, saving the exponent and the squared error (r^2) to a central result table and a plot of each combination.

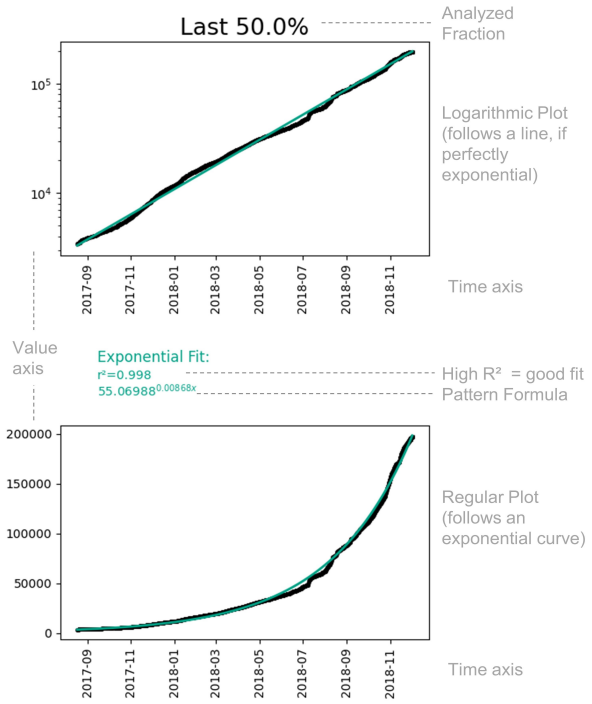

Figure 2. Plot interpretation

With n rows, k date columns and m numerical columns in a data set, the complexity of this algorithm is $O(m * n * k)$. For each pattern a plot as depicted in Figure 2 is saved for analysis. The plot shows the same data for each date-value combination in a regular plot (bottom) and logarithmic plot (top). If the data fits an exponential function perfectly, the logarithmic plot shows a straight line in the logarithmic plot. The data points are drawn in black, the fitted exponential function is added in green. Additionally, the r² and the formula are provided. Finally, the top label shows if the pattern was found in the full time range, in the last 50% (as is the case in the example), or the last 25%.

Table 1. Runtime Pattern Finding Algorithm

Hardware	Size	RAM	CPU	Time
Local Machine	180 GB (11,612 files)	16 GB	4	80h
GCP E2-Standard-4 Machine	180 GB (11,612 files)	16 GB	4	36h

Performance Evaluation: When run on the full set of 180 GB of Kaggle data with 11,612 files), it takes 80 hours on a commodity machine (Intel i5 processor with 4 cores and 8 GB RAM) to complete (Table 1). The e2-standard-4 Linux machine in Google Cloud only requires 36 hours. The difference in time can be attributed to GUI and background processes running on the windows machine compared to the Linux machine. This run-time does not suffice for applying this algorithm to a corporate data lake or repository.

4.2. Scaled Distributed Architecture

To make that algorithm scalable to petabytes of data, the master node distributes the data sets across multiple worker nodes in the scaled architecture we propose in Figure 3. If there are n data tables and m machines, then each machine would run Algorithm 1 on $\frac{n}{m}$ chunk of data sets. So the overall time reduces to $\frac{n}{m} * t$. The steps are represented in Algorithm 2.

Algorithm 2 Distributed Potential Finder Algorithm

K = Total number of Kaggle data sets
N = Total number of workers
Start with machine 1, i.e, i = 1
D_i = data set to be processed by $Worker_i$
S_i = Number of the data set in D_i
for $j = 1, 2, \ldots K$ **do**
  if $S_i \leq \frac{K}{N}$ **then**
    Add Data set j to D_i, S_i += 1
  else i += 1
  end if
end for
for $i = 1, 2, \ldots N$ **do** parallel
  for $j = 1, 2, \ldots S_i$ **do**
    Open the file in read mode, read the first line
    Write the first line to $sample\_file[j]$
    nline = Line count of that file,.
    for $k = 1, 2, \ldots nline$ **do**
      Generate a divisor, d = $int(\frac{nline}{5000})$
      if remainder $(\frac{k}{d})$ == 0 **then**
        Write line[j] to $sample\_file[i]$
      end if
    end for
  end for
  Start Algorithm 1 for samples of data set D_i
  R_i = Results of Algorithm 1 for samples of D_i
  Report R_i
end for

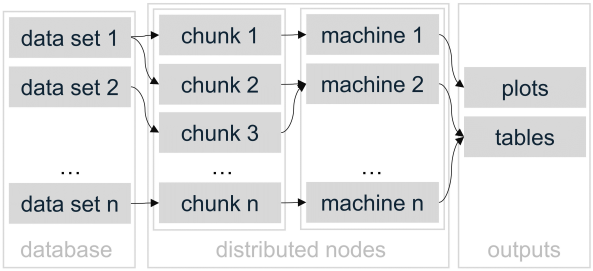

Figure 3. Distributed Architecture

Byte-Wise Sampling: As reading and processing large

files slows down the process substantially, we introduce byte-wise sampling for each file. The goal is to extract a fixed amount of randomly selected rows (set to max. 5000, or the total number of rows if the data table has fewer rows), to keep the run-through time per table low. Byte-wise means, that we do not read the full file before sampling (like standard libraries like pandas in Python do), but rather identify the correct rows and read only them by using the steps: 1) get n_{rows}; 2) get $num_{line}\ mod(\frac{n_{rows}}{5000})$; 3) read only the selected num_{lines} and skip all other rows without reading them.

Performance Evaluation: The performance evaluation results are shown in Table 2. By using the sampling combined with the distribution approach to 4 nodes, a speedup by 98% could be realized compared to a local machine for 180 GB of data. If more speed is required, it can be achieved by introducing more workers reducing the run-through time to the desired level.

Table 2. Distributed Pattern Finding Algorithm.

Machine	Hardware	Size	Time
M1	E2-4	45 GB (4,100 files)	1.4 h
M2	E2-4	45 GB (3,200 files)	1.3 h
M3	E2-4	45 GB (1,850 files)	1.2 h
M4	E2-4	45 GB (2,500 files)	1.3 h

4.3. Similarity Network

Being able to individually inform data owners that their data contains exponential patterns is already possible with the algorithms in Section 4.1 and Section 4.2. However, we want to go one step further and suggest new connections between the data owners working in different contexts and fields. The goal is to create innovation networks based on the patterns found in the data. To achieve this, the similarity between the found patterns is used to connect the sources and their data owners to innovation networks.

In the distributed pattern algorithm 'Potential Finder' a table of all found patterns is collected, providing information about the exponent, the analyzed fraction (0.25, 0.5, 1), the r^2 and the measure for even distribution (see Section 5.1). Additionally, we use the Python library TSFresh to calculate additional columns describing the found patterns automatically in a numeric way. From that, a nodes table including all found patterns and their properties is written.

All numeric columns are then used in a cosine similarity calculation, providing a similarity value for all combinations of exponential patterns in an adjacency matrix. This matrix is stacked to get all combinations in a tabluar form and filtered for similarity strength of $>$ 0.5, as we only want to keep strong similarities in the network. The resulting table is stored as an edge table.

Cytoscape [30] is used for calculating graph centrality indices and for creating the visual representations in Section 5.2.

Performance Evaluation: The resulting patterns found from the Kaggle test data set were filtered to 757 patterns (by filtering for an $r^2 > 0.95$ and $continuous\_distribution\_index < 0.05$). After filtering the network edges for similarity strength > 0.5, 132,901 edge combinations remain for the network. The whole network is computed in 3 seconds on a local machine.

4.4. Cluster Analysis

The last step on the way to enable the data owners in this innovation network to work together on exponential patterns is to reduce the individual communication effort among data owners by clustering them in communities with similar exponential patterns. For that, all computed properties from the previous steps plus the node connectivity indicators from the network are used as a basis for a K-Means clustering [31]. The resulting clusters are provided in Section 5.2.

Performance Evaluation: Classifying the 757 patterns and and their 132,901 weighted connections took 2 seconds on a local machine.

5. Results

5.1. Patterns Identified by the Distributed Algorithm

After initial loading and filtering, 11,612 data sets have been extracted from the 180 GB Kaggle data. To reflect that a pattern can be a recent one or a long-term one, each Date-Numeric combination was checked three times: taking 1) the full time range, 2) only the last 50%, 3) only the last 25% into evaluation.

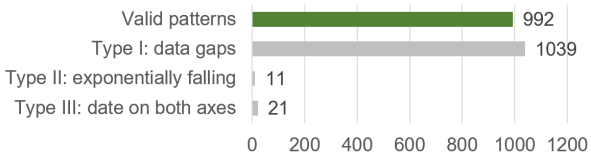

Figure 4. Classes of Patterns

To focus on valuable patterns, a r^2-threshold of 0.8 is used, resulting in $2,063$ patterns remaining. The lower

the r, the less the data resemble an exponential pattern, and the less likely it is that small investments or actions targeting that pattern can have a major impact in the future (because the process described by the data does not follow a 'law' but is acting randomly). On manual inspection of these exponential patterns, we found that many of them had high r² and growth parameter, but still did not qualify as relevant exponential pattern. By reviewing them manually (taking 18 hours), we identified 48% (992) patterns to look valid. We classified the misfits into three types:

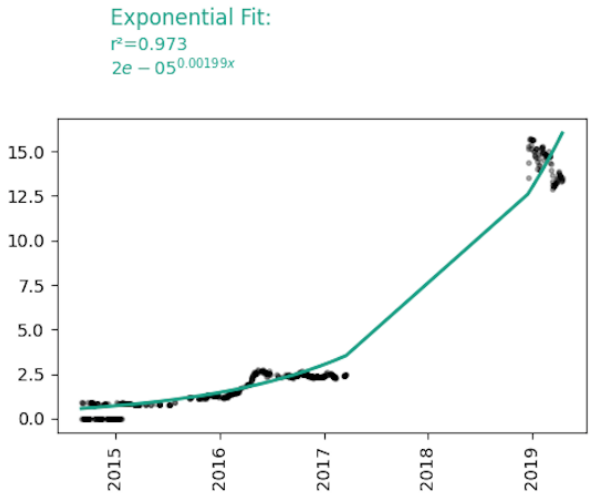

Figure 5. Misfit Type I: Missing Data

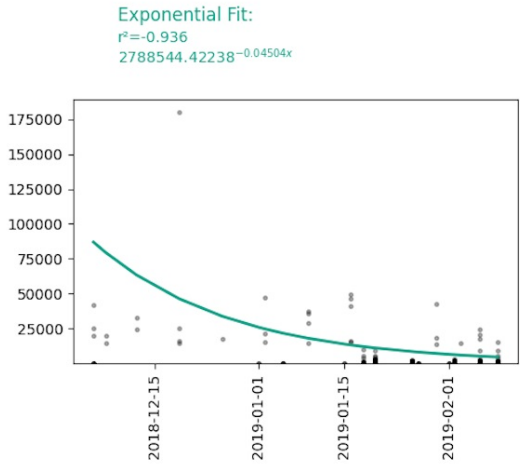

Figure 6. Misfit Type II: Falling pattern

Type I misfits had overall exponential nature but also systematic gaps in the data distribution over time. Figure 5 is a pattern found in a data set 'FIFA19-Ultimate Team players' for the combination of sales_date and xbox_last value column. It shows an exponential pattern but there is a systematic gap in the data points for which the pattern is drawn.

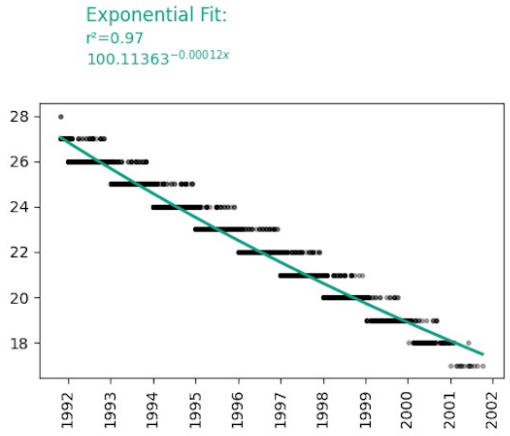

Figure 7. Misfit Type III: Date column on both axes

Type II misfits are exponentially decreasing in nature. Figure 6 shows an example of a pattern found in the sales of the xbox gaming setup. As we are interested in exponential growth to maximize business potential, such patterns are dropped.

Type III misfits have date columns on both axes. Figure 7 shows the columns 'age' and 'year of birth' from the FIFA-19_ultimate_players data set.

Additional Filters addressing misfit Type I-III:

- **Type I**: To filter out patterns with missing data, we introduce a $continuity\_index$ over time: For each pattern, We divide its data points into 50 equal intervals and then count the number of buckets without data points. To normalize to a value between 0 and 1 we divide by 50. The higher this index, the more segments of this pattern have no values across time, the worse we consider this function to fit to an exponential function. As we want no big gaps to be existent, set the threshold to 0.05, so only 5% of the 50 buckets are allowed to be without a data point. This drops 813 misfit patterns.

- **Type II**: We remove the data points that have negative slope of the exponential coefficient. This drops 11 misfits.

- **Type III**: We check if the column name of the numeric column is date or any of the data in the column can be converted to date type and consequently remove it. This drops 18 misfits.

Found Exponential Patterns: An example of a found pattern is shown in Figure 8. The pattern represents the opening point measurement of Dollar Tree (DLTR) in the stock market plotted against the time period for

which it was recorded. This exponential pattern could be interesting for a business that is involved in consulting or some venture fund organization who are constantly looking for such growing business domains so as to invest in those to make a successful investment. All found patterns are provided as overview table and as plot.

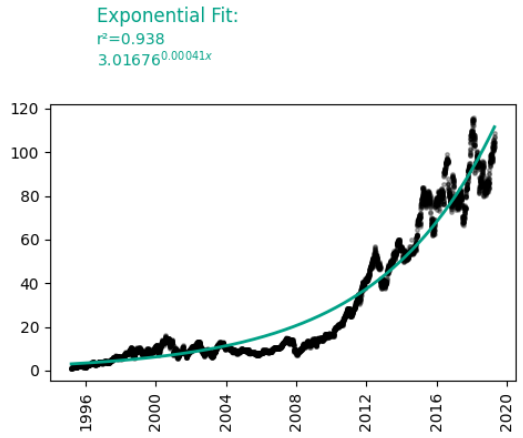

Figure 8. Sample Exponential Pattern Extracted

To focus on high quality patterns, we increase the r²-threshold to 0.90, resulting in 751 patterns remaining. Figure 9 shows the number of all found patterns by Data Owner and data table, which gives also an overview over the fields and topics of the found patterns.

Data Owner	Tablename	# Found Patterns
qks1lver	amex-nyse-nasdaq-stock-histories	640
pablote	nba-enhanced-stats	49
solorzano	rave-dr5-gaia-dr2-consolidated	15
chicago	chicago-311-service-requests	8
abeserra	wikia-census	5
new-york-city	ny-2015-street-tree-census-tree-data	4
new-york-city	ny-housing-maintenance-code	3
morriswongch	kaggle-datasets	2
new-york-state	nys-active-corporations-beginning-1800	2
pankrzysiu	weather-archive-jena	2
piterfm	ukraine-deputies	2
san-francisco	sf-curb-ramps	2
sohier	calcofi	2
tsiaras	uk-road-safety-accidents-and-vehicles	2
viniciusbbizarri	sorteiosmegasena	2
yingwurenjian	chicago-divvy-bicycle-sharing-data	2
city-of-seattle	seattle-crime-stats	1
cms	cms-state-summary-of-outpatient-charge-data	1
fivethirtyeight	fivethirtyeight-most-common-name-dataset	1
fvcoppen	solarpanelspower	1
lucian18	mpi-on-regions	1
new-york-city	ny-handyman-work-order-hwo-charges	1
new-york-state	nys-child-care-regulated-programs	1
snapcrack	all-the-news	1
yoghurtpatil	311-service-requests-pitt	1

Figure 9. Overview Found Patterns per Data Owner and Table

5.2. Innovation Network and Subgroup Clustering

The resulting network with all remaining 751 patterns is depicted in Figure 10. In the graph, the layout is computed based on the edge weight (=similarity index) with the Prefuse Force Directed Layout in Cytoscape [30]. Found patterns with strong connection (=similarity index) are drawn close together, the distance between the others is maximized. The bigger a point is, the higher its R^2 is. The color codifies the cluster provided by the cluster algorithm K-Means euclidean distance. The clustering algorithm used the elbow method [32] to find the optimal number of clusters. In our example and data, this led to 4 different clusters depicted in purple, yellow, green and black. The highlighted patterns in red are all uploaded from a single Kaggle Data Owner named "solorzano".

Forming of Innovation Groups: In the exemplary data set downloaded from Kaggle, 282 Kaggle users are present as data owners. In the network generated, patterns were found for 22 data owners. With the similarity network we enable each data owner to check for similar patterns and contact the other owner. Each pattern has on average 255 similar patterns connected, so the data owners would have to communicate too much to engage in innovation. The clustering allows to group similar patterns in groups. The data owner highlighted in red in Figure 10 is, for example, part of clusters one and four, and would only have to engage with 9 other data owners in two innovation groups. The clustering, therefore, reduces the individual effort and automatically distributes innovation effort into innovation groups to foster crowd innovation.

6. Discussion

6.1. Usefulness of Exponential Patterns for Business Innovation

We focus on exponential patterns in this paper for the starting point for the Innovation Network Creation Framework, as we see the most immediate benefits in those patterns. They are easy to fit, easy to interpret, follow a clear upward trend over time (as we only analyze time-related exponential patterns), they are asked for in startup pitches and allow for predictions of the further development of the process or behaviour represented in the data. For example the 640 stock market patterns in 4 can be used for informing mid-term investment strategies. Another example would be developing a product to increase efficiency for the city of Chicago and New York service requests (as they

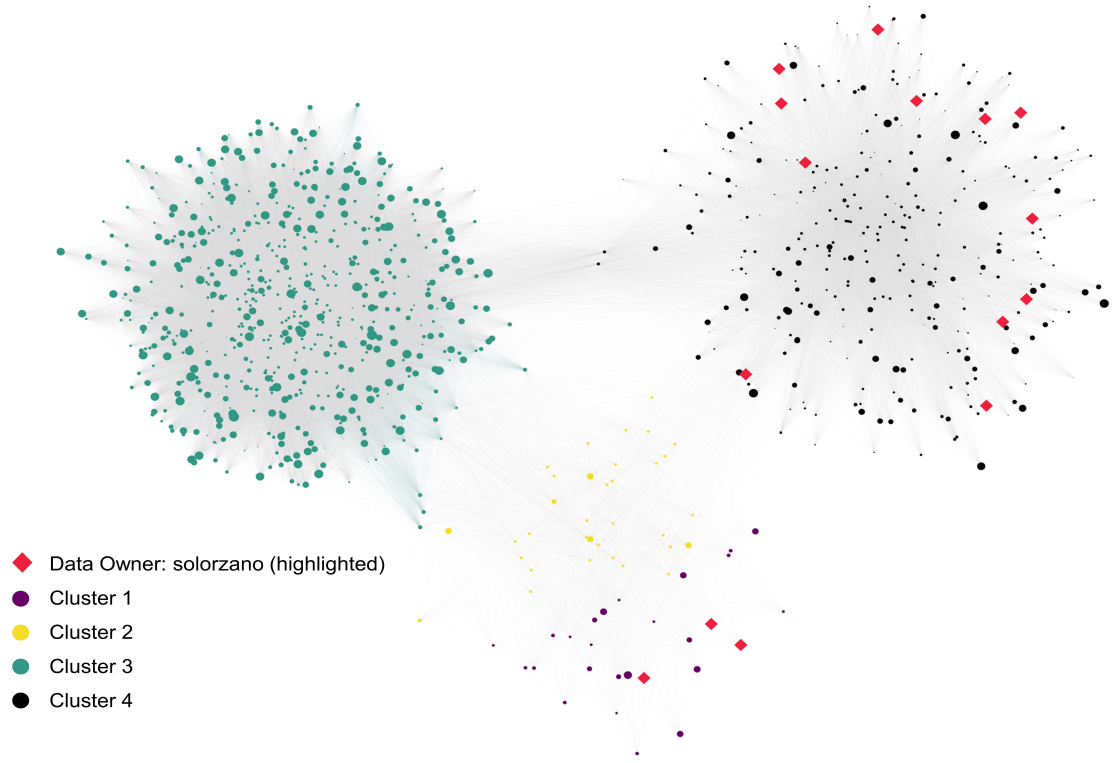

Figure 10. Innovation Network Around Exponential Patterns With Clustering

grow exponentially a solution targeting at those will bring more ROI than building a service for a shrinking niche target group) These properties make them an interesting starting point for what we envision as a framework for multiple types of patterns in the future. They are, however, not without shortcomings. From virus spreads in biology we know, that in nature no exponential growth is lasting forever. For example, Covid-19 outbreaks had exponential growth rates only for a limited time, because the nature of exponential virus spreading was known and detected and the governments and society globally took costly and often drastic measures to prevent the further exponential spread. As soon as the target population is saturated or measures against the exponential growth are actively taken, exponential growth will fade into an logistic function. Moore's law and digital services as well as the growth of data worldwide are examples of exponential growth processes, where no active countermeasures (like in virology) are taken and the break point seems to be in much further future. The major challenge we see in working with exponential patterns to build your business model on, is to realize if you are still in growth mode or if the target saturation is reached. As described in Section 7 we aim to supplement this analyses with logistic pattern detection and a stronger focus on integrating meta-information on the patterns found as the next steps. This more holistic view will help to judge the life cycle position of a pattern and inform business analysts, innovators and entrepreneurs if investing in the process behind a found pattern is worth the investment. Additionally, our approach allows to scan for exponential growth behaviour in public data to inform actively taking counter measures against unwanted areas of exponential growth (like virology).

6.2. Prevalence of Exponential Patterns in Data

Our research shows that exponential patterns can be found regularly in public tabular data sets. The developed library works with all data tables with date information (i.e. minimum one numeric column which is parsed as date). Consequently, time-series data, panel data, cross-section data is analyzed column-wise and related exponential patterns are extracted. In 180 GB of public data, we found 424 exponential patterns of different data owners. On average, that amounts to 2.34 exponential patterns in 1 GB of tabular data. Having found the pattern does not replace actively innovating around it. Our approach to scale this usage across a newly suggested innovation network in sub-user groups

enables crowd-innovation around exponential patterns. The more that data are collected and stored publicly and privately, the more finding such patterns on low cost monitoring would allow for broad-scale crowd innovation.

6.3. Experimental Setups and Cost Analysis

To demonstrate that the framework can be realized minimum-cost, we conducted our experiments only on local machines and on Google cloud platform using the $300 free trial credits. In our experiments, we used e2-standard-4 instances which have 4 virtual vCPUs and 16 GB Memory costing $0.134 per hour. We also used cloud storage buckets on the google cloud platform for storing the data sets downloaded from Kaggle. The cost for a cloud storage bucket is $0.02 per GB per month. By using high performance instances the total computing time could be reduced further significantly. Overall, the cost for executing the sampled distributed algorithm on 180 GB data was calculated to be around $0.804 and the same for 100 TB data would be $400.57 (projected). Also, there would be cost associated with storing the data in Google cloud bucket. So, for storing 180 GB data for a month it would cost $3.60 and the same for 100 TB data would cost $2,048 (projected). As documented in Sections 4.3 and 4.4, the network creation and clustering caused only marginal resources and can be done on a local machine. Summing up the efforts for all framework steps: for the price of a Mac Book Pro you can search for 100TB of data and identify roughly 250k exponential patterns.

6.4. Scalability

The developed algorithm allows to scale the exponential search across worker nodes. The exemplary application was done on 220 GB of public data from Kaggle (resulting in 180 GB of usable data). This is far away from industrial data lakes and repositories used, for example, in autonomous driving. It is however, closer to business data table data sizes with several tabular data sets rather than sensor or media data. The proposed architecture gives the user flexibility in terms of choosing between more time (and low cost) or more speed (with higher short time cost), as the computation can be scaled to petabytes of data as described in the cost analysis. With that, both public and private organizations are enabled to search for such patterns. The nature of the identification process also does not require immediate results, this algorithm can also run silently in the back and process existing data sources requiring minimal maintenance.

6.5. Limitations

To show that this approach is possible to scale at minimum cost, we only used very limited freely available resources. Given more computation power and pattern evaluation capacity, the approach could be scaled and extended to much larger volumes.

Design Science Research aims to create and evaluate in iterative cycles. The quantitative behavioural evaluation of the innovation network to form was excluded as a research topic to focus the research more on the intersection between data science, engineering and organizational science. However, we see this as the starting point for more depth in all three of those research streams in follow-up work.

7. Conclusion and Outlook

Data grows exponentially. Our paper and the presented framework enables public and private organizations to identify exponential growth within that expanding sea of potentials with a scaled architecture at minimum cost ($4.50 for scanning 1 TB). It allows Information systems and organizational researchers to not just analyze innovation networks, but to actively create them around similar patterns in data, bridging knowledge, discipline and organizational boundaries and connect data owners with investors, data scientists and entrepreneurs worldwide.

We see the extension of the architecture to also detect additional patterns as the next step. Logistic patterns would also allow to identify processes which have been growing exponentially but have arrived in a steady state. Cyclical patterns would allow to react on seasonality. Similarly to the newly introduced continuity index in Section 5.1, further focus on feature engineering and the supplementation of meta information on the data analyzed can further improve the similarity linkage. Another step forward could be the application of the framework to private data lakes and repositories to study and create innovation networks in the private sector. Given more analysis capacity, diving deeper into the patterns found to ideate business models in cooperation with startups or exponential thinking education programs would allow to further scale the approach. Empirical research could guide and evaluate the forming and development of these new innovation networks to study the behaviours and success factors of this anarchic type of innovation network creation.

References

[1] M. E. Porter, *Competitive advantage: creating and sustaining superior performance*. New York : London: Free Press ; Collier Macmillan, 1985.

[2] R. Normann and R. Ramírez, "From value chain to value constellation: Designing interactive strategy.," vol. 71, no. 4, pp. 65–77, 1993.

[3] K. Lyytinen, Y. Yoo, and R. J. Boland Jr, "Digital product innovation within four classes of innovation networks," *Information Systems Journal*, vol. 26, no. 1, pp. 47–75, 2016.

[4] L. Albinsson, M. Lind, and O. Forsgren, "Co-design: an approach to border crossing, network innovation," *Expanding the knowledge economy: issues, applications, case studies*, vol. 4, no. Part 2, pp. 977–983, 2007.

[5] C. M. Cunningham, P., "Expanding the knowledge economy; issues, applications, case studies.," *Reference Research Book News, Ringgold, Inc*, vol. 23, no. 4, pp. 1039–1045, 2007.

[6] P. Galison *et al.*, *Image and logic: A material culture of microphysics*. University of Chicago Press, 1997.

[7] Y. Yoo, O. Henfridsson, and K. Lyytinen, "Research commentary — the new organizing logic of digital innovation: An agenda for information systems research," *Information Systems Research*, vol. 21, pp. 724–735, Dec. 2010.

[8] X. Zhao, "A scientometric review of global bim research: Analysis and visualization," *Automation in Construction*, vol. 80, pp. 37–47, 2017.

[9] J. Pollack and D. Adler, "Emergent trends and passing fads in project management research: A scientometric analysis of changes in the field," *International Journal of Project Management*, vol. 33, no. 1, pp. 236–248, 2015.

[10] H. Lietz and M. Riechert, "Science dynamics: normalized growth curves, sharpe ratios, and scaling exponents," in *Proceedings of the 14th international society of scientometrics and informetrics conference*, vol. 2, pp. 1566–1577, 2013.

[11] Y. Fang, "Visualizing the structure and the evolving of digital medicine: a scientometrics review," *Scientometrics*, vol. 105, no. 1, pp. 5–21, 2015.

[12] C. Chen, R. Dubin, and M. C. Kim, "Emerging trends and new developments in regenerative medicine: a scientometric update (2000–2014)," *Expert opinion on biological therapy*, vol. 14, no. 9, pp. 1295–1317, 2014.

[13] W. Zhou, J. Chen, and Y. Huang, "Co-citation analysis and burst detection on financial bubbles with scientometrics approach," *Economic research-Ekonomska istraživanja*, vol. 32, no. 1, pp. 2310–2328, 2019.

[14] R. M. Gandia, F. Antonialli, B. H. Cavazza, A. M. Neto, D. A. d. Lima, J. Y. Sugano, I. Nicolai, and A. L. Zambalde, "Autonomous vehicles: scientometric and bibliometric review," *Transport reviews*, vol. 39, no. 1, pp. 9–28, 2019.

[15] L. Bettencourt, D. Kaiser, J. Kaur, C. Castillo-Chavez, and D. Wojick, "Population modeling of the emergence and development of scientific fields," *Scientometrics*, vol. 75, no. 3, pp. 495–518, 2008.

[16] R. Arévalo, J. García, F. Guijarro, and A. Peris, "A dynamic trading rule based on filtered flag pattern recognition for stock market price forecasting," *Expert Systems with Applications*, vol. 81, pp. 177–192, 2017.

[17] F. Wenig, P. Klanatsky, C. Heschl, C. Mateis, and N. Dejan, "Exponential pattern recognition for deriving air change rates from co 2 data," in *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pp. 1507–1512, IEEE, 2017.

[18] H. Chesbrough, "Open innovation results: Going beyond the hype and getting down to business," vol. 1, no. 1, pp. 86–116, 2020.

[19] R. Mashelkar, "Exponential technology, industry 4.0 and future of jobs in india," *Review of Market Integration*, vol. 10, no. 2, pp. 138–157, 2018.

[20] C. Phelps, R. Heidl, and A. Wadhwa, "Knowledge, networks, and knowledge networks: A review and research agenda," *Journal of management*, vol. 38, no. 4, pp. 1115–1166, 2012.

[21] J. Singh, "Collaborative networks as determinants of knowledge diffusion patterns," *Management science*, vol. 51, no. 5, pp. 756–770, 2005.

[22] L. Fleming, S. Mingo, and D. Chen, "Collaborative brokerage, generative creativity, and creative success," *Administrative science quarterly*, vol. 52, no. 3, pp. 443–475, 2007.

[23] J. E. Perry-Smith and P. V. Mannucci, "From creativity to innovation: The social network drivers of the four phases of the idea journey," *Academy of Management Review*, vol. 42, no. 1, pp. 53–79, 2017.

[24] R. Huggins and P. Thompson, "Entrepreneurship, innovation and regional growth: a network theory," *Small Business Economics*, vol. 45, no. 1, pp. 103–128, 2015.

[25] S.-W. Kwon, E. Rondi, D. Z. Levin, A. De Massis, and D. J. Brass, "Network brokerage: An integrative review and future research agenda," *Journal of Management*, vol. 46, no. 6, pp. 1092–1120, 2020.

[26] H. A. Simon, "The sciences of the artificial," pp. 111–138, 2019.

[27] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS quarterly*, pp. 75–105, 2004.

[28] S. Gregor and A. R. Hevner, "Positioning and presenting design science research for maximum impact," *MIS quarterly*, pp. 337–355, 2013.

[29] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *Journal of management information systems*, vol. 24, no. 3, pp. 45–77, 2007.

[30] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, "Cytoscape: a software environment for integrated models of biomolecular interaction networks," *Genome research*, vol. 13, no. 11, pp. 2498–2504, 2003.

[31] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[32] A. Coates and A. Y. Ng, "Learning Feature Representations with K-Means," in *Neural Networks: Tricks of the Trade: Second Edition* (G. Montavon, G. B. Orr, and K.-R. Müller, eds.), Lecture Notes in Computer Science, pp. 561–580, Berlin, Heidelberg: Springer, 2012.